

Chapter Title: Dwarven Epitaphs: Procedural Histories in Dwarf Fortress

Chapter Author(s): Stephanie Boluk and Patrick LeMieux

Book Title: Comparative Textual Media

Book Subtitle: Transforming the Humanities in the Postprint Era

Book Editor(s): N. KATHERINE HAYLES and JESSICA PRESSMAN

Published by: University of Minnesota Press

Stable URL: <https://www.jstor.org/stable/10.5749/j.ctt5hjjtq.11>

JSTOR is a not-for-profit service that helps scholars, researchers, and students discover, use, and build upon a wide range of content in a trusted digital archive. We use information technology and tools to increase productivity and facilitate new forms of scholarship. For more information about JSTOR, please contact support@jstor.org.

Your use of the JSTOR archive indicates your acceptance of the Terms & Conditions of Use, available at <https://about.jstor.org/terms>



University of Minnesota Press is collaborating with JSTOR to digitize, preserve and extend access to *Comparative Textual Media*

JSTOR

6

Dwarven Epitaphs: Procedural Histories in *Dwarf Fortress*

Stephanie Boluk and Patrick LeMieux

WITH THE RISE of digital inscription technologies, the history of the twenty-first century will not be written by human hands alone. From surveillance systems to medical databases to networked computers, automated processes of information and exchange are already encoding an alternative history of humanity and coauthoring the stories of the next century. Smart phones log global position, online advertisements count clicks per capita, search engines index content (and “personalize” results) in real time, and **automatic backup applications record the state of your hard drive so you can “revisit your [computer] as it appeared in the past”** (Apple 2012). Nowhere are these invisible informatics more deeply felt than through the algorithmic operations driving contemporary financial markets. At 2:45 p.m. on May 6, 2010, subsecond stock exchanges between networked trading systems on Wall Street produced an unprecedented, system-wide “flash crash,” which caused the Dow Jones to plummet a thousand points, only to recover just moments later. Now known as the “Crash of 2:45,” this micro-crisis inspired a group of physicists at the University of Miami to recover and analyze 18,520 ultrafast black swan events to study the long-term economic impact of algorithmic trading on global financial systems post-2006 (Johnson et al. 2012). Considering the massive amounts of data being produced at speeds and scales that exceed the limits of human phenomenology, future historians will find themselves in a similar situation, studying the history of the twenty-first century not in terms of human-legible days, weeks, months, and years but in machine-scale microseconds.

Like researchers applying physics to flash crashes, communities

have formed to study the complex dynamics governing play within contemporary videogames. Of the games that foreground their operational qualities and create play from the engagement with complex systems, one stands out: a text-based computer game by Tarn and Zach Adams called *Dwarf Fortress* (2002–). The game begins by generating an expansive, highly detailed world in which up to one thousand years of natural and social history are dynamically produced. Despite its minimal textual interface, the process of generating this history weighs heavily on the central processing units of most computers. The millions of events logged during world generation are granular enough not only to correspond to the history of the gamespace represented on-screen but also to ultimately historicize the processor cycles of the computer itself. In rejecting the conventions of standard game design and traditional narrative storytelling, *Dwarf Fortress* produces textual inscriptions that not only mark one horizon of human experience but recall forms of historical writing that depart from human-centered and teleological models of history, such as the annal, chronicle, and calendar. In response to (and despite) the difficulty of actually engaging with the game's deanthropomorphized, speculative terrain, a community has formed over the past decade dedicated to exploring, documenting, and translating the results of *Dwarf Fortress* into texts we call *Dwarven Epitaphs*. *Dwarven Epitaphs* are monuments of human play and a powerful form of comparative textual media produced amid a computational wilderness. These postmortem accounts of play navigate the interstices of fan fiction, fan translation, and fan archiving not only to expand the field of play beyond the boundaries of software but also to explore the intersection of making and critique. Whereas *Dwarf Fortress* embraces the granularity of computation over the specularity of graphical realism, delighting in the strange, formal logics of the computer, *Dwarven Epitaphs* convert world generator to story generator and mark the dissonant registers produced between human operators and nonhuman operations.

STRIKE THE EARTH!

Before playing *Dwarf Fortress*, one must first write the history of the world. There are no preset level designs or narrative scenarios

authored in advance. Instead, *Dwarf Fortress* offers players dozens of menus filled with detailed parameters for generating over one thousand years of natural and social history. Basic options include the size and age of the world, the composition of the soil, the character of the wildlife, the number of civilizations expected to develop, and even the probability of “forgotten beasts” roaming the underworld. Once these parameters are set, the application’s smallish, 640×300 pixel window begins to whirl with activity as everything from badger boars to bauxite boulders emerge from combinations of simple mechanics (see Figure 6.1).

World generation begins with a single seed: a pseudo-random number that acts as a kernel initiating *Dwarf Fortress*’s dynamic systems. What starts as straightforward elevation variants and temperature settings births atmospheric pressure systems that condense to form rivers, lakes, mineral formations, and, finally, vegetation. From there, ecosystems develop supporting different types of forests, foliage, fruits, and flowers. Various species of insects, rodents, predators, and prey settle each virtual environment alongside dwarves, elves, and humans. Mountains move and seasons change as civilizations spread and roads stretch across the 40×40 character grid of twinkling tokens. The typographic characters seen here are mimetic: a dark green “♠” could represent a particularly dense forest, whereas a teal “⊥” might denote a tributary flowing into a larger river. Zooming in for a closer look, these characters change their meaning. As Jonah Weiner (2011) writes in an article for the *New York Times*: “a normal person looks at ♠§dg and sees gibberish, but the *Dwarf Fortress* initiate sees a tense tableau: a dog leashed to a tree, about to be mauled by a goblin.” Despite its constrained poetics and modest visual scale, *Dwarf Fortress*’s complex physics simulations manage to stress the processing capacity of most computers, slowing other activity to a crawl while an otherwise unassuming pocket planet slowly develops.

After world generation is complete, the player is given the option between three distinct play styles: “Legends,” “Adventure,” and “Fortress.” By far the most popular way to play and the game’s namesake, Fortress mode begins by randomly assembling a company of seven dwarves, who embark to a site of the player’s choosing in the hope of founding a new dwarven outpost. During

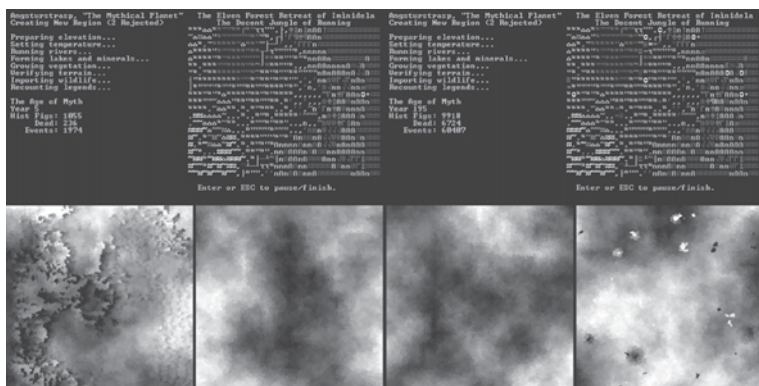


FIGURE 6.1. World generation (above) can produce over one thousand years of nonlinear history based on dynamics such as rainfall, drainage, savagery, and volcanism (below).

the painstaking preparation process, each of the seven dwarves is outfitted with tools and trained in specific professions. These jobs range from the staples of agrarian fantasy, like mining, woodcutting, fighting, fishing, and brewing, to the arcane and often absurd beekeeping, lye making, siege engineering, glass blowing, record keeping, pump operating, and even small animal dissection. After choosing an assortment of household supplies, domesticated animals, and alcoholic beverages (which the dwarves require “to get through the working day”), the player’s new party of ☺’s are plunked into the middle of an entropic environment. Their first task? To construct an architecture resistant to both the forces of nature and the volatile nature of the dwarves themselves.

At this point, the game becomes a mixture of city building, military strategy, and physics simulation. In *Dwarf Fortress*, the player can mark mountains to be mined, trees to be chopped, plants to be gathered, and architecture to be built via the designations menu (see Figure 6.2). After constructing workshops, further designations allow players to direct dwarves to build furniture, plant seeds, switch levers, and, of course, brew beer. Unlike most simulation, city-building, and strategy games, however, there is no guarantee that any one dwarf will respond exactly to the player’s demands.¹ If the carpenter has not built enough barrels to store the beer being brewed, the brewer will simply quit his job and find something

else to do. Dwarves form elaborate chains of interdependence that are always at risk of being interrupted by bad weather, bad attitudes, or even a stray badger boar or bauxite boulder. **As the game grows in complexity, the truly autonomous nature of the dwarves intensifies, as what could be called a “dwarven culture” begins to spontaneously emerge.** While a couple holes in the ground are adequate to suit the needs of seven dwarves, seventy dwarves will not peacefully coexist without organized labor, social hierarchy, clean living conditions, and a constant flow of alcohol. Even with barracks, kitchens, workshops, and stockpiles in constant upkeep, there is always the possibility of a single disgruntled dwarf setting off a chain reaction of violence or “tantrum spiral”—their fragile psyches only rivaled by the fragility of their civilization.

The relationship between player and dwarf is only strained further by the game’s minimal visual design and frustratingly counterintuitive graphic user interface (GUI). Despite the availability of both two- and three-dimensional GUIs, many of *Dwarf Fortress*’s players continue to interface with encoded representations in the form of simple matrices of multicolor type. More specifically, *Dwarf Fortress* uses a set of slightly modified code page 437 (CP437) characters in sixteen different colors to represent its world, a throwback to the textual interface of MS-DOS on the original IBM PC. Though not strictly text based (the individual glyphs are actually bitmap textures rendered in OpenGL and can be exchanged for alternative tile sets [see Figure 6.3]), *Dwarf Fortress* is clearly programmed in the idiom of text-based gaming, adding techniques like artificial intelligence (AI) and physics simulation to the lineage of games such as *Colossal Cave Adventure* (1975–6) and especially *Rogue* (1980).² More visually complex rendering techniques, such as overlapping, transparency, and three-dimensional positioning, are eschewed in favor of simpler effects.³ Though many contemporary games are built on the mechanic and thematic innovations of these textual predecessors, *Dwarf Fortress* employs the enhanced processing power of contemporary home computers to push not pixels but processes.

With the constant uncertainty of both the dwarven AI and the stone age GUI, building a dwarf fortress often seems like an exercise in futility. Just learning the game’s controls and interface—a

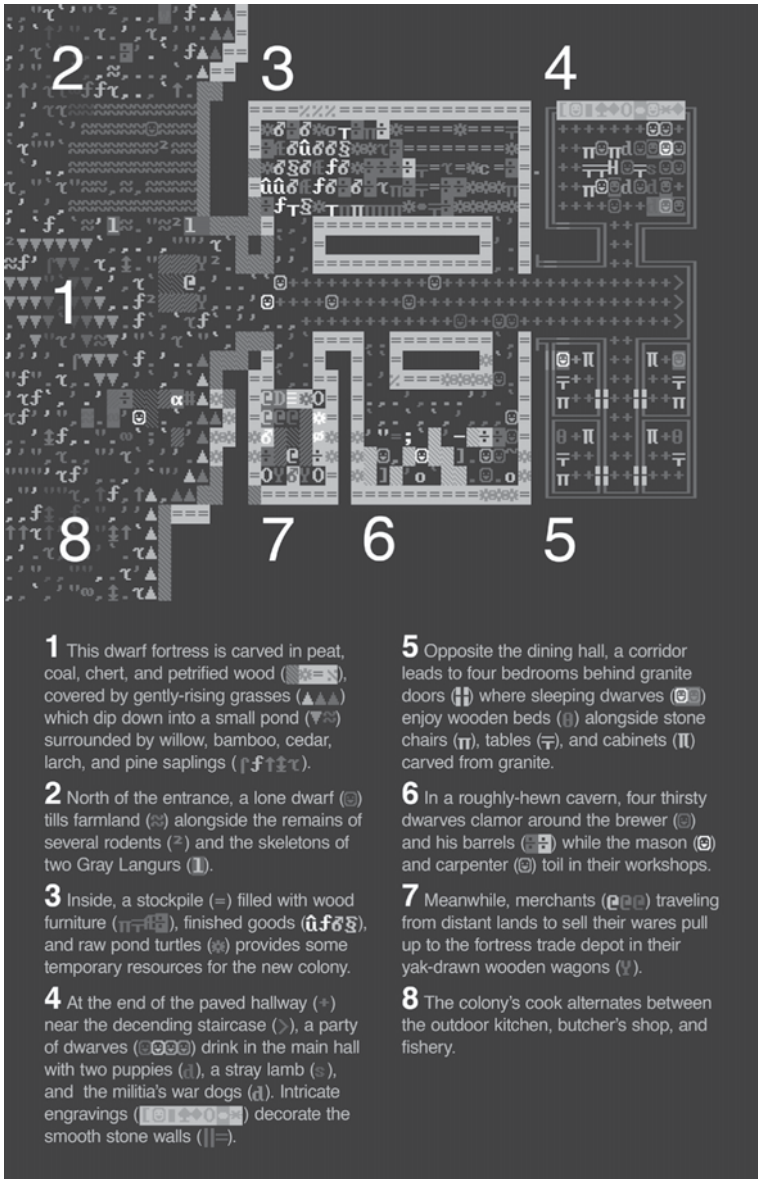


FIGURE 6.2. A cross-section of a growing dwarf fortress reveals about thirty dwarves living with their pets and livestock among bustling workshops and brimming stockpiles.

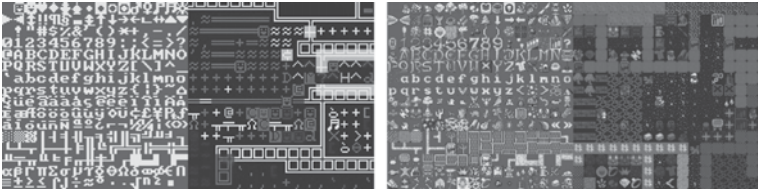


FIGURE 6.3. *Dwarf Fortress*'s standard tile set (left) compared with “Gold Plated” by Russian community member DrD_AVEL (right).

labyrinth of nested, text-based menus—is a serious undertaking that has inspired hundreds of hours of fan-made video tutorials, thousands of articles on the community-run “MagmaWiki,” and even publications like Peter Tyson’s (2012) *Getting Started with Dwarf Fortress: Learn to Play the Most Complex Video Game Ever Made*. Alongside these substantial usability issues, it is in no way clear *how* to build a dwarf fortress or even *what* a dwarf fortress is. Though a player may use the game’s menus to designate a given space as “bedroom” or “dining room,” there are no strict requirements regarding their architecture outside of constructing a bed or table. With no clear-cut goals, only self-motivated players continue to tinker with the system after their first exposure, and this general lack of direction opens the question of whether *Dwarf Fortress* is a game at all. In the end, players decide for themselves whether it is toy or tool, sandbox or simulation.

Uncertainty in terms of how to play is at least partly due to the particularities of the Adamses’ production process. For the past ten years, *Dwarf Fortress* has existed as a constant work in progress, an alpha build (currently 0.34.11) promising future versions beyond the horizon of 1.0, a release that the Adamses joke could take decades. Before proper beta testing begins, an “alpha” designation implies the phase of development in which a videogame is first playable but by no means complete. Although, in the case of *Dwarf Fortress*, alpha status allows the Adamses to continue developing new dynamics for their simulation, the indefinitely incomplete project results in frequent updates, instability, and a general lack of refinement typically resolved by in-house playtesting. Owing to the combination of this seemingly infinite alpha status with the game’s capricious mechanics, bare-bones interface, and sheer

difficulty, Greg Costikyan (2011) concludes that “*Dwarf Fortress* is a game from an alternate universe.” He imagines a history of videogames in which “there never was a revolution in computer graphics,” although “Moore’s Law . . . progressed just as it has in our own [timeline].” For Costikyan, *Dwarf Fortress* conjures a parallel reality in which “[Chris] Crawford’s 1980s claim that ‘process intensity’ rather than ‘data intensity’ was the future of games” turned out to be true. *Dwarf Fortress* challenges the past three decades of game design so completely that Costikyan feels compelled to situate it within another world entirely because “no one in his right mind would have created it in our own.”

In celebration of the game’s unique, almost utopian problematics, the community ethos has become “failing is fun,” and players have made it a practice to narrate their experiences, translating the inscrutable actions that take place in *Dwarf Fortress* into more legible accounts of play. Players exhume the remains of ancient civilizations and build Dwarven Epitaphs commemorating past plays and future failures. In *Archaeologies of the Future*, Fredric Jameson (2005, xiii) writes that “at best Utopia can serve the negative purpose of making us more aware of our mental and ideological imprisonment . . . and that therefore the best Utopias are those that fail most comprehensively.” The ludic and narrative failures of this “game from another world” disrupt the logic of capital and produce an alternative history of videogames dictated not to production schedules, release dates, intellectual property rights, and executable code but to play.

NARRATIVE BOOKENDS

Dwarf Fortress begins and ends with stories. Narratives serve as both the inspiration for Adamses’ software as well as a frequent result of playing with their system. The game’s design can be traced to a series of short stories written by Zach Adams, a student of ancient history and an amateur writer with a penchant for bite- (or byte-) sized genre fiction. To devise new game mechanics for *Dwarf Fortress*, the brothers carefully analyze Zach’s short stories in an attempt to uncover the underlying dynamics driving the logic of the fantasy genre, or at least its most recurring settings and plotlines (Adams and Adams 2006–11). After their collaborative

analysis, Tarn sorts each idea into card catalogs, which rank *Dwarf Fortress*'s design goals according to their plausibility and urgency. Working from mid-afternoon into the early morning on a daily basis, Tarn integrates these rules into the larger system of algorithmic processes running *Dwarf Fortress*. Needless to say, the resulting simulation never unfolds completely as expected. The experience of playing *Dwarf Fortress* can swing wildly between frustration and delight as the emergent mechanics and occult interface obfuscate any immediate understanding of the simulation. Because of its arduous gameplay and outright difficulty, the inevitable fall of each dwarven fortress is often followed by an afterword. In a rare case in which history is written by the losers, players return *Dwarf Fortress* to the textual format from which the Adams brothers began by narrativizing their experience on blogs and forums. They exchange short stories, illustrations, comics, schematics, analysis, and even modifications of Tarn and Zach's system. New ways to play are constantly being invented as these player-produced narratives complete the long circuit of translation and codevelopment between human and computer to become Dwarven Epitaphs.

Before building fortresses and erecting epitaphs, the Adams brothers mined Zach's short stories to unearth the game's mechanics. In one of his entries titled "Battlefield Lunch," Zach details a range of activities surrounding the carrion left in the wake of a large battle. The main figure of the tale is a mysterious, "barely humanoid" monster that shares a gore-ridden field with the "widows of fallen soldiers searching for enemies that were still alive so they could torture them with their long knives" (Adams and Adams 2006). Based on this short episode, in which a lone hero is tripped by a dying foe, only to become the beast's "freshest meal . . . in weeks," the Adams brothers discover several behaviors not yet simulated in *Dwarf Fortress*. Among their notes are the following:

- large scavengers eating the dead
- entity hatred can manifest through acts of torture on those that can't defend themselves
- wrestling: tripping people, becoming impaled on ground objects (Adams and Adams 2006)

By analyzing the narrative to extract its implicit tropes and conventions, the Adamases attempt to translate the rules of the storyworld into the algorithmic processes running *Dwarf Fortress*. The hope is that, by identifying and encoding patterns such as revenge, scavenging, and grappling, narrative events akin to those described in “Battlefield Lunch” will propagate widely across the simulation. Thus *Dwarf Fortress* is not so much a digital-born artifact as it is “born translated.” In his essay in this volume (chapter 11), John Zuern appropriates Rebecca Walkowitz’s concept of “born translated” to argue for the “affinity between comparative literature and digital literary studies,” and *Dwarf Fortress* stands as an excellent example of an object that cannot be regarded as a stable piece of software but rather as a process of intertwined production and play that is constantly engaged in a series of textual translations and remediations.

This translation process, however, is deepened when each of Zach’s original stories is encoded into a computational environment operating under an entirely alternative set of temporal, spatial, and causal logics. The result of distilling and translating genre fiction into game mechanics does not produce infinite, ready-made fantasy stories that mirror Zach’s writing. Instead, *Dwarf Fortress* generates an even more interesting, destabilized spool of diachronic ticker tape documenting a thousand years in the span of a few seconds. The computer produces the stories of processor cycles as much as cyclopes and refresh rates as much as ratmen. In one game set across a thousand-year time scale, *Dwarf Fortress*’s world generation will typically produce about fifty thousand noteworthy characters participating in over half a million events in thousands of locations. When confronted with these sprawling catalogs, stylistically reminiscent of only the barest of narrative forms—the calendar, the chronology, and the chronicle—the player is left to wonder how, for example, does a gathering cold front in the early autumn of the year 128 impact her activities over a thousand years later? Why document the construction of the market Kux Ungod, “The Cup of Gallows,” in the first year of the world? What was so important about the incident of the stolen rutile ring, spirited from Elevated Dungeon in Blizzardwitch by the kobold Lrakildus? Considering these are just three of the

millions of randomly generated plot points produced during one instance of *Dwarf Fortress*'s world generation (and the average player will typically generate multiple worlds before deciding to settle on one scenario), **a comprehensive survey of this material is impossible.** Given an entire lifetime, no human could ever exhaust the content of this book of sand.

After *Dwarf Fortress* simulates its stream of serial events, players interact with these macro- and microhistories to translate the processes of a computer back into legible, narrative forms. Tim Dennee's quirky and beautifully rendered comics *Oilfurnace* and *Bronzemurder* depict the bizarre domino effects leading up to the destruction of more than one dwarven civilization (see Figure 6.4). Inspired by the events of specific play sessions, these blocky, paper cutout-style renderings perfectly capture both the human error and dwarven eccentricities that make "losing fun" in *Dwarf Fortress*. Another example of translation can be seen in the painstaking research and attempts to narrativize the remarkable history of the dwarf Tholtig Cryptbrain, a statistical anomaly discovered in *Dwarf Fortress*'s Legends mode and the sole survivor of a century-and-a-half-long war in which she claimed 2,341 unique kills. Tholtig's story is a death-defying feat not of skill but of chance and probability. Finally, and even more puzzling, "The Design for a Dwarven Computer" is a computer engineering project that attempts to render the computational processes of *Dwarf Fortress* in terms of the dwarves themselves and has been dubbed the game's first "megastructure." The schematics for this mechanical computer depict expansive fields of pumps, levers, and switches that blanket most of *Dwarf Fortress*'s in-game environment and, given the speed of water and willfulness of the dwarves, exhaust the labor and life cycle of an entire civilization to process a few clock cycles of this prehistoric computer. These players examine the unexpected and often inscrutable operations of *Dwarf Fortress* by relocating millions of in-game deaths from the register of computational statistics back into human-legible accounts of play. Instead of forfeiting at the first sign of adversity (or anomaly), the Dwarven Epitaphs of *Oilfurnace*, Tholtig Cryptbrain, and the Dwarven Computer transform the end of the game into an endgame in itself—moving beyond the software

to engage in a form of comparative textual media by translating the actions of the game into walkthroughs, schematics, and novelizations.

The most popular player-created history of *Dwarf Fortress* is the collaboratively authored tale of “Boatmurdered” (2006–7). Originally penned on *The Something Awful Forums*, “Boatmurdered” documents a “succession game” in which different players took turns overseeing and recording the growth (and inevitable decay) of a single fortress (aptly named Boatmurdered) over the course of several in-game years. The player-authors produced a narrative framework from the game’s procedural paradoxes. For example, in this particular playthrough, herds of hostile elephants wandered the jungles near Boatmurdered and became surprisingly disruptive to the development of dwarven civilization. The elephants, like many of the other creatures in this ecosystem, exhibit behaviors noted in Zach’s short stories. For example, take the idea listed earlier that “entity hatred can manifest through acts of torture on those that can’t defend themselves.” One could speculate that these behaviors emerge in the elephants outside Boatmurdered as the players witnessed them “slowly torturing a war dog stuck outside” the walls of the mountain fortress (Various 2006–7). Here the ♠§dg from Weiner’s earlier example transforms into an even stranger (and perhaps more unsettling) ♠§dE.

While this example demonstrates the relationship between Zach’s narratives and Tarn’s programming, the combination and layering of *Dwarf Fortress*’s complex dynamics take on a life of their own and produce unpredictable effects that do not always align with traditional forms of narrative and psychological causality. In another entry from the Boatmurdered saga, the bellicose elephants slay a dwarf who has wandered too far from the protection of the fortress. To one player’s horror, an unfortunate loop propagates as, one by one, individual dwarves scurry from the safety of their mountain home onto the bloody field to loot and recover the corpses of their fallen countrymen. By exhibiting this form of scavenging behavior, one of the character traits explicitly identified in “Battlefield Lunch,” almost twenty dwarves fall victim to this computational trap before the player intervened, writing (Various 2006–7),



FIGURE 6.4. Excerpts from the webcomic *Oilfurnace*, by Tim Dennee, remind players that “losing is fun.”

Harsher fucking measures are necessary. I install two front doors and lock them. No one gets in or out. Boatmurdered is closed until further notice. . . . This basically stops the elephant problem for now as a temporary measure. . . . We lost about 20 dwarves to this debacle.

This unexpected effect is the aggregate of multiple game mechanics. By encoding Zach’s stories into *Dwarf Fortress*, Tarn must have placed a higher priority on the routines driving each dwarf to acquire misplaced goods or stockpile corpses in designated zones over the functions accounting for the presence of danger. As they scamper across the battlefield, the artificial intelligence of each dwarf ignores the fact that the trampled bodies he intends to pick-pocket are the product of those elephants still in close proximity.

In writing their collaborative history of *Boatmurdered*, the succession of players inevitably anthropomorphize computational processes by assigning psychological behavior to the dwarves, describing them as irrationally “bound and determined to march to their deaths” (Various 2006–7). The legends created in and around

Dwarf Fortress embrace the peculiar narrative effects emerging from the extended feedback loop between authorial intent and the experience of play. But exactly what happens between these two narrative bookends? Given the scale of the simulation and the number of rules interacting with each other, there is no way for the player to fully anticipate the emergent effects of dwarf and environment alike. Rather than lament this feature (which might even be construed as bugs or glitches in any other game), the possibility of inexplicable events is one of *Dwarf Fortress*'s most celebrated qualities. The pleasure in being unable to entirely foresee how the system will overwhelm the player is one of the reasons why, as Tim Dennee quips, "losing is fun." The ephemeral and distinctive character of each gameplay session is what inspires players to mark particularly dramatic losses through the production of Dwarven Epitaphs like *Oilfurnace* and "Boatmurdered." These postmortem accounts of *Dwarf Fortress* narrativize the human experience of computational processes by appropriating the language of death to express not only the unexpected expiration of their dwarves but also the end of a traditional notion of play. Dwarven Epitaphs do not commemorate the fantasy of mastery in which one "beats" the game but instead appropriate death as a means to celebrate ephemeral moments of play in a system where "winning" is no longer a measurement of value.

THE EMERGENCE OF A DWARVEN COMPUTER

Videogames, unlike games of dice, cards, chess, or physical sports, conflate the rules of a game with the mechanics governing the environment in which the game is played. For example, nowhere in the rules of baseball is gravity defined, whereas in procedural games like *Dwarf Fortress*, the explicit authoring of forces such as gravity, weather systems, fluid simulation, and geophysics takes the place of traditional rule sets. Rules are social configurations, pacts between players not to peek or move outside invisible boundaries. As Bernard Suits ([1978] 2005, 10) argues in his philosophical treatise on games titled *The Grasshopper: Games, Life, and Utopia*, "rules prohibit more efficient in favor of less efficient means, and . . . playing a game is the voluntary attempt to overcome

unnecessary obstacles.” Mechanics, conversely, are involuntary and necessary; there is no choice but to work within the constraints of these systems. Whereas rules can be broken, mechanics cannot be turned off. In this sense, there is no cheating in *Dwarf Fortress*. Exploits that rely on glitches or unplanned programming infelicities are still working within the logic of the software. From this perspective, direct modifications of the game’s source code and each software update result in the production of a new and ontologically distinct game. Once *Dwarf Fortress* is operational, anything goes—from completely passive pause states to micromanaging the movements of a dwarven militia, there is no wrong way to play. This game does not have rules, only mechanics. Thus it could be argued that *Dwarf Fortress* is not a game at all but rather an ontological experiment.

Instead of suggesting guidelines from which to play, like “three strikes and you’re out,” *Dwarf Fortress* offers the ball and bat. For the community, the game functions as a tool set or platform from which to playfully experiment, author narratives, and invent new games. So though it can be called a physics simulation, a city-building game, and a story generator, the Adamses’ software might feel more at home in the popular sandbox genre of computer software. Sandbox games are designed to allow players to experiment with a robust set of game mechanics without predicating play on any one scripted sequence of actions, events, and goals. Ranging from physics toys featuring particle and fluid dynamics to open-world, digital playgrounds where players can roam free, the most successful sandbox games model simple mechanics that combine to yield dynamic operations. The unexpected ludic consequences and extemporaneous narratives produced by these mechanics enable and encourage players to invent their own games within games—to build metagames like the Dwarven Epitaphs previously discussed. It is no surprise, then, that Tarn and Zach Adams invest in the concept of emergence as a core design philosophy within *Dwarf Fortress*. Emergent events seem nondeterministic but are caused by a small set of simple mechanics. When set in relation to one another, these mechanics become greater than the sum of their parts. In this sense, the Adamses are building Dwarf Physics more than *Dwarf Fortress*.

By simulating a physical world via emergent mechanics, *Dwarf Fortress* establishes an isomorphic relationship between geological processes and computation in a way that recalls Manuel De Landa's (1997) work on Gilles Deleuze and Félix Guattari's philosophy of emergence in *A Thousand Years of Nonlinear History*. In this "neomaterialist" reading of Deleuze and Guattari's "10,000 B.C.: The Geology of Morals (Who Does the Earth Think It Is?)," De Landa claims that "sedimentary rocks, species and social classes (and other institutionalized hierarchies) are all historical constructions, the product of definite structure-generating processes" (62). This kind of history, however, is not arbitrarily constrained to man-made writing but considers the movements of geological strata (alongside those of genes, memes, norms, and money) as forms of inscription in and of themselves. Recalling *Dwarf Fortress*'s computational landscapes, De Landa describes the accumulation of sedimentary rocks like sandstone or limestone as an emergent process in which "rivers [act] as veritable hydraulic computers (or at least, sorting machines) . . . [and sedimentation] is carried out by certain substances dissolved in water (such as silica or hematite in the case of sandstones) which penetrate the sediment through the pores between pebbles" (60). De Landa (1995) extends this homology between computation and sedimentation not only to pebbles drifting along the bottom of a river but also to the genetic drift of biomass across the surface of the planet. Despite this capacious account of organic matter in *A Thousand Years of Nonlinear History*, De Landa returns to what he calls a "basic geological spirit, a philosophical stance which rejects ideas of progress not only in human history but in all natural history as well." "Living creatures," he writes, "are in no way 'better' than rocks. . . . Indeed, in a nonlinear world in which the same basic processes of self-organization take place in the mineral, organic and cultural spheres, perhaps rocks hold some of the keys to understand sedimentary humanity, igneous humanity and all their mixtures." And what better than a dwarf to represent sedimentary and igneous humanity?

The flattened moral landscape of *Dwarf Fortress* dramatizes De Landa's nonhuman geological philosophy. If historical inscriptions underwrite the processes of nature, then there is a kind of code in

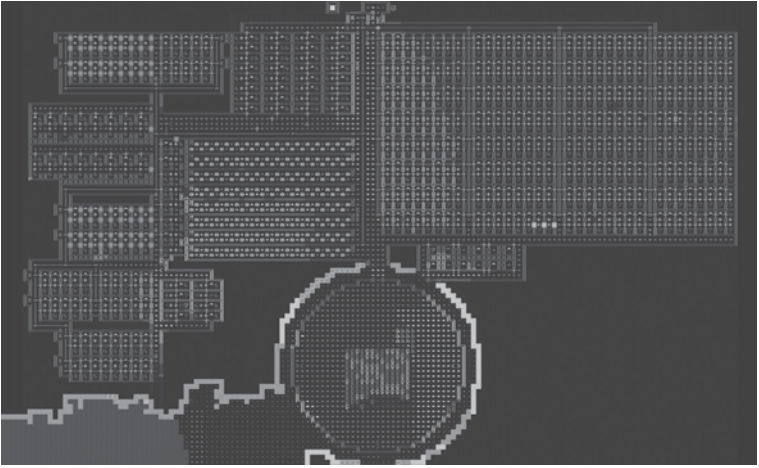


FIGURE 6.5. Jonathan Ng's "Design for a Dwarven Computer" envelops the landscape within *Dwarf Fortress*.

the soil. *Dwarf Fortress* expresses this logic by explicitly tying the activity of mining to computing. To dig is to code, and each dwarf is a data miner, a cog within a larger allegory of computational processes. "Computation, as defined in computer science, is a description of a regularly patterned machine operation," writes Terrance Deacon (2012, 95), and "in this respect, the workings of [a] desktop computer, or any other computing machine, are just the electronic equivalents of levers, pendulums, springs, and gears, interacting to enable changes in one part of the mechanism to produce changes in another part." This entwining of *Dwarf Fortress*'s geological processes with the mechanical and computational is best expressed through one of the most virtuosic "mega constructions" ever undertaken in *Dwarf Fortress*: the Design for the Dwarven Computer (see Figure 6.5).

Using *Dwarf Fortress*'s robust mechanics to produce circuitry, power sensors, registers, memory units, machine instructions, and arithmetic logic unit (ALU) architecture, Jonathan Ng has designed a fully functioning, universal computer within the game itself (Ganapati 2010). At first only appearing as a perverse inversion of scale, the full absurdity of the project is made clear once the digital logics of a computer are mechanized by canals, pumps,

and locks, moving water through mountains and maintained by dwarven militias for months as they slowly churn through the logical processes and computational tasks of a mainframe computer. Echoing De Landa, the rivers of *Dwarf Fortress* quite literally act as “veritable hydraulic computers,” as the rise and fall of simulated water levels become storage devices for bit-wise operations.

In his essay “Reading *exquisite_code*: Critical Code Studies of Literature” (chapter 12 in this volume), Mark Marino writes that critical code studies “takes the analytic approaches of software studies and applies them to the source code.” In 2012, Marino convened a working group in which *Dwarf Fortress* was discussed as an unconventional platform or “strange IDE” (integrated development environment) for designing projects like the Dwarven Computer, Dwarven Calculators, and even a Dwarven Game of Life based on John Horton Conway’s classic example of emergence (Buluk and LeMieux 2012). Though building a fully functional ALU within the game itself cannot be considered “source code,” the tools available to the player are flexible enough to implement many of the fundamental operations that make coding possible: Boolean variables, conditional statements, and even rudimentary operations like addition, subtraction, multiplication, and division. As Nick Montfort (2010) pointed out in a brief blog post, the Design for the Dwarven Computer is adapted from *The Elements of Computing Systems* (2005), a popular textbook that computer engineering students use to practice building CPUs.⁴

Although functional, the Dwarven Computer operates within the geological time scale of the game. A single calculation could take years of in-game time and require a multitude of mechanisms spanning acres of in-game geography. Moreover, the Dwarven Computer is in constant threat of being corrupted by the unpredictable and often slovenly behavior of the dwarves who maintain its mechanism (not to mention goblin infestations and even environmental processes like evaporation and leaking in the slowly pumping logic gates). By transposing the typical speed of computer processing into a geological register, these in-game computers juxtapose sub-human microtemporalities with superhuman macrotemporalities to conflate two limit cases of human experience. *Dwarf Fortress* enables a computer to not only remake the world but also remake

itself within that world. A potentially endless loop of creation and emergence oscillates between these geological and computational registers, while the human player is reduced to the position of helpless bystander, the flickering symbols of a dwarven wilderness spinning on and off at a subsecond or century-long time scale.

A HISTORY OF THE WORLD

“As history has infiltrated physics,” De Landa (1997, 15) writes, “we must now allow physics to infiltrate human history.” This is the position from which *Dwarf Fortress* commences: by inscribing a speculative history that begins, almost biblically, with the birth of the world and the creation of a thousand years of nonlinear history. Within this time span, civilizations form and legends are written featuring only a fraction of the events created during world generation (some hundred thousand entries). In *Dwarf Fortress*’s Legends mode, the player has access to text logs detailing the procedural history of every geographical formation, architectural construction, civilization, and historical figure. It is possible, for example, to pull up the historical records of a dwarf who lived and died in the subsecond cycle of a computer processor. The microtemporality of this emergent environment signals the horizon of human phenomenology. When framed against the scale of Earth’s geological time, human lives flicker on and off as quickly as the billions of binary operations that drive each dwarf.

Thousands upon thousands of seemingly insignificant details are indexed from a much larger set of processes. Each of these tiny transactions is then parsed into queryable databases and assigned textual signifiers. Cobbled together from the Adamses own fictional etymologies and organized in the form of a chronology, computer processes are given names. The result is a textual archive that reads more like *The Silmarillion* than *Lord of the Rings*.⁵ The irony of *Dwarf Fortress*’s microtemporal history is the way its procedural capaciousness is thrown into relief by its minimal textual output and interface. Taking a random sample:

- In the early spring of 1036 Fiviirafe, “The Furious Assault” occurred.

- In the early spring of 1037, The Pelts of Stabbing defeated The Complex Councils of The Snarling Empires and pillaged Touchbottle.
- In the early spring of 1037, The Oracular Realms accepted an offer of peace from The Static Lyrics.
- In 1042, the human Nani Hopplait became the master of The Nations of Gravel.
- In the early spring of 1043, The Rampage of the wolf Perad Crewedtrifle in Beachstop occurred.

The historical texts of *Dwarf Fortress* seem to exemplify what Noah Wardrip-Fruin (2009, 16) calls the “*Tale-Spin* effect.”⁶ *Tale-Spin* is an early story-generation system invented in the mid-1970s by James Meehan while he was studying at Yale. Wardrip-Fruin argues that the simplicity of *Tale-Spin*’s textual output obscures the complex processes required to simulate a navigable space, to set interactive objects within the space, and model the psychological processes of its inhabitants. The *Tale-Spin* effect can be seen in the stilted, curt language and sometimes unintentionally surreal events produced by *Dwarf Fortress*’s complex dynamics. Despite the attempt to describe the movements of several artificially intelligent agents in the preceding example, the historical sequence appears strange and arbitrary, with no transitional expressions or conjunctive adverbs to tie together the sentences. There is a hidden complexity that operates according to the formal consequences of computation rather than the narrative logic of an author or reader.

If the *Tale-Spin* effect is produced when a deceptively straightforward interface obfuscates some underlying, operational complexity, then the “*Dwarf Fortress* effect” may describe the deliberate implementation of the *Tale-Spin* effect to produce ludic, cognitive, or phenomenological defamiliarization. Usability, in this case, is contingent on inscrutability. The Adamses deploy an anachronistic textual interface that draws on earlier modes of historical writing to extend the homology between the geological and computational processes of *Dwarf Fortress* to the concept of history itself. *Dwarf Fortress*’s linguistic forms of historical inscription in Legends mode bear a striking formal and thematic resemblance to early forms of writing, such as the medieval annal and chronicle. In *The Content*

of the Form, Hayden White (1987) defines the annal as a collection of historical events listed year by year. The annal is distinguished from the chronicle, which also takes on the form of a list but is organized around a strong central subject (e.g., a church's history). Both forms lack the narrative causality that White attributes to contemporary historical writing, and for this reason, both have been regarded as somehow incomplete or underdeveloped modes of historical representation. By contrast, White argues that the annal and the chronicle are not ideologically primitive forms of inscription but instead embody a fully developed but radically different philosophical worldview. He proposes that these forms of medieval writing have been misrecognized in much the same way that it is easy to misinterpret the output of *Dwarf Fortress*. Like *Dwarf Fortress*, the annal deprivileges the place of the human within a larger cosmological landscape.

Compare the previous excerpt from *Dwarf Fortress* to the *Annal of Saint Gall*, which serves as White's principal case study (as quoted in White 1987, 6–7):

- 709. Hard winter. Duke Gottfried died.
- 710. Hard year and deficient in crops.
- 711. Hard year and deficient in crops.
- 712. Flood everywhere.
- 713.
- 714. Pippin, mayor of the palace, died.
- 715. 716. 717.
- 718. Charles devastated the Saxon with great destruction.
- 719.
- 720. Charles fought against the Saxons.
- 721. Theudo drove the Saracens out of Aquitaine.
- 722. Great crops.
- 723.
- 724.
- 725. Saracens came for the first time.
- 726.
- 727.
- 728.
- 729.

730.

731. Blessed Bede, the presbyter, died.

732. Charles fought against the Saracens at Poitiers on Saturday.

733.

734.

One could describe these narrative forms as having suffered from a kind of literary *Tale-Spin* effect, having been denigrated as incomplete historical productions compared with more familiar and narratively rich modes of contemporary historical writing. While the paralinguistic features of modern typography have homogenized the way in which writing is interpreted, in chapter 5 of this volume, William A. Johnson notes that the legibility of historical forms of writing often depended not on information contained within the text itself but on the particularities of historical reading practices. Where to pause, breathe, separate words, toggle between various columns, raise or lower one's voice, or speed up or slow down were conventions passed on from teacher to student and not encoded within the bookroll itself. Reading was a social practice embedded within a much larger cultural and medial context. And while reading texts from *any* era always entails an act of interpretation, this fact is compounded when learning to read historical documents with alternative paralinguistic features. The same situation applies to learning the rules and linguistic codes of *Dwarf Fortress*'s eccentric textual interface. The Adamases' choice to depict the in-game historical events as overwhelming in terms of scale yet narratively sparse is not simply a product of the immense technical constraints and challenges that govern the development of a complex procedural storytelling system. By simulating a mode of historical inscription from which modern readers are estranged, *Dwarf Fortress* produces an allegory for the relationship between human and nonhuman. Both the microtemporality of computer process and the macrotemporality of geological phenomena trace their own kinds of nonlinear, nonhuman histories within different scales of silica. These times and scales, both big and small, macroscopic and microscopic, operate outside the phenomenal capacity of any human reader to produce a history without subjects.

Both Johnson and White take contrasting approaches toward

the absence of information in early writing. Whereas Johnson attributes a rich and lively sociotechnical context through which bookrolls circulated and on which their meaning was dependent, White understands the gaps and narrative disjunctures as symbolic of an alternative philosophy of history. **White attempts to recuperate the annal and chronicle forms by arguing that they are not “imperfect histories” but rather “possible conceptions of historical reality, conceptions that are alternatives to, rather than failed anticipations of, the fully realized historical discourse that the modern historical form is supposed to embody”** (White 1987, 6).

For White, the form of the annal rejects Hegelian grand narratives of historical progress and presents a world indifferent to human struggle. Both the medieval annal and the chronologies of *Dwarf Fortress*’s Legends mode follow a narrative model that is nonteleological and that deprivileges the place of the human within historical time. White not only reads specific genres of writing as nonteleological, but argues for a reading of the history of history which is itself nonprogressivist. In this way, White resists historiographic narratives that suggest historical writing culminates in familiar styles of contemporary forms. But beyond the formal similarities between White’s medieval texts and the Adamses’ software, there are also significant thematic resonances as they catalog related types of events. White notes how the annal “immediately locates us in a culture hovering on the brink of dissolution, a society of radical scarcity, a world of human groups threatened by death, devastation, flood, and famine” (7). Similarly, the inhabitants of *Dwarf Fortress* are at constant risk of being overrun by the indifferent computational wilderness in which they are placed. Although both *Dwarf Fortress* and *The Annal of Saint Gall* depict a society in a state of perpetual crisis, there are few reasons explicitly given in either that explain why events occurred. White writes,

Why “Charles fought against the Saxons” remains as unexplained as why one year yielded “great crops” and another produced “flood everywhere.” Social events are apparently as incomprehensible as natural events. They seem to have the same order of importance or unimportance. They seem merely to have occurred, and their importance seems to be

indistinguishable from the fact that they were recorded. In fact, it seems *their importance consists in nothing other than their having been recorded.* (8; emphasis added)

In *Dwarf Fortress*, the geological, biological, and social elements of the game exist as similarly incomprehensible forces and function on a flattened ontological plane. Events in *Dwarf Fortress* are recorded simply because they *happened*. Echoing White's remarks on the annal, their importance consists in "nothing other than their having been recorded." White emphasizes that annalistic forms of historical inscription make no reference to a "system of merely human morality or law" (14) that is comparable to the computational logic that drives *Dwarf Fortress*, a game in which the histories of merchants, mountains, kings, and cats are documented with equal orders of magnitude and the experience of the reader or player appears only as an afterthought. White attributes the seeming neutrality of the annal to being a product of a theistic cosmology: "since everything that happened did so apparently in accordance with the divine will, it is sufficient simply to note its happening, to register it under the appropriate 'year of the Lord'" (13).

Supporting the comparison between medieval cosmology and flattened ontology—as depicted in the Legend mode's sprawling databases, which log computational processes in the form of historical inscriptions—players resort to assigning metaphysical and supernatural explanations to unexpected in-game phenomena. On the occasion of a particularly surprising or illogical game mechanic (some of which the human player or programmer might characterize as bugs or glitches yet that are in no way ontologically distinct from other game mechanics), it is not uncommon for players to narrate their dwarves experiencing a divine act of God or some form of supernatural magic, neither of which is currently implemented in Tarn and Zach's system. Instead, the machinic processes of *Dwarf Fortress* are cast in a role similar to that of an indifferent, absent God—a kind of *deus ex machina* that is strikingly distinct from the version Aristotle critiqued in his *Poetics*. Rather than the unmotivated yet just-in-time divine intervention of a user-friendly deity that responds to prayers and

solves queries, the god-in-the-machine of *Dwarf Fortress* is a black box, a force that infinitely recedes from view and from whom the human remains forever and inexorably estranged.

DWARVEN EPITAPHS

Although operating from different ends of the philosophical spectrum, both White's narratology and De Landa's realism argue for a nonteleological model of history that rejects narrative coherence and the logic of progress in favor of a model of discontinuous and emergent complexity. The rise of electronic forms of inscription applies further pressure to the traditional concept of history. As theories of history have been deeply wedded to forms of chirographic and print writing, it is no coincidence that hyperbolic claims about the "end of history" have been made in an era in which print has come to the end of its dominance.⁷ Digital technologies are not only reconfiguring historical models, but perhaps the perception of digital media as endlessly new and its implied lack of historicity are the result of lingering print-centric assumptions. In any case, new forms of historical inscription are now being automated on an unprecedented scale. Networked videogames log every keystroke while a Google Document records where, when, and by whom each revision was made. Smart phones report GPS coordinates and generate maps of each carrier's geographical movements, while databases, digital archives, and cloud computing support the storage of thousands of petabytes of data. Digital technologies are writing a microtemporal history of dynamic processes that include, but are not limited to, the movements of humanity. Despite this extrafunctional, machinic history appending each and every technological inscription, we continue to simultaneously ignore *and* overestimate our position as the subject of history. The history of the dwarf, however, throws human history into radical relief.

By rendering historical forms of narrative on a microtemporal scale, *Dwarf Fortress* simulates contemporary models of historical inscription. The game operates according to the same automatic and serialized logic as telephone records, bank statements, GPS systems, and e-mail exchanges. By attempting to narrativize records of processor cycles and refresh rates, players produce Dwarven

Epitaphs. Dwarven Epitaphs are memorials, monuments of critical play—a productive, practical play that generates metagames. These player-designed games created in, on, around, and about other games resist and operate outside the ontology of the Adamses' procedural world. They break the rules and make losing fun. Metagames relocate abstract, autonomous software into human spheres of play and teach us to recognize the actions of the nonhuman agents with which we constantly collaborate in all aspects of contemporary life. In this sense, dwarves live within us, around us, and without us—in vaccines and antibiotics, in mechanical and computational systems, and in the geological and cosmological happenings of the universe. The role of the human, then, is not to play videogames but to produce metagames. Dwarven Epitaphs historicizing the strange interactions amid incomprehensible ontological territories. And though we can speculate on the existence of dwarves and how deeply they affect our lives, we can never experience the radically distinct ontological spaces of the digital, the temporal, and the cosmological. The lives of dwarves are a mystery, but players continue to memorialize their history on blogs, forums, and wikis. Dwarves must die for this game to be fun.

NOTES

- 1 In *Dwarf Fortress*, the role of the player is not to control each individual dwarf, as with most real-time strategy (RTS) games starting with *Dune II* (1992) and the original *Warcraft* series (1994–2003). Though the RTS genre shares the same bird's- or god's-eye perspective with *Dwarf Fortress*, its gameplay relies almost entirely on harvesting, building, defending, and attacking with units that respond to the player's mouse clicks in real time. Dwarves, conversely, have a mind of their own. They are more similar to the autonomous agents populating city-building games like *SimCity* (1994) than the space marines of military-strategy games such as *StarCraft* (1998). However, in contrast to *SimCity* and most simulation games' depiction of humans as rational agents whose population size correlates directly to player progress, dwarven behavior is idiosyncratic at best and seemingly random at worst. The player's task, then, is to attempt to

- curb the disruptive capacities of the dwarves by designating tasks that will keep them busy, comfortable, and full of enough food and drink to remain stable.
- 2 Interactive fiction like Will Crowther and Don Woods's *Colossal Cave Adventure* and Infocom's *Zork Trilogy* (1980–82) allowed players to interact with environmental descriptions and spatial riddles through a command-line interface able to register simple expressions such as “look up,” “walk north,” and the iconic “get lamp.” In the text-based dungeon crawlers following *Rogue*, players guide a typographic glyph (often an anthropomorphic “@” sign) through randomly generated mazes of periods, pipes, and other forms of punctuation. Both textual descriptions and text-based representations are implemented in the Adamses' game.
 - 3 E.g., if two or more characters occupy the same tile in *Dwarf Fortress*, their cohabitation is visualized through a simple animation that cycles through each entity's symbol in a slow blink. Somewhat misleadingly, this same method is used to show status updates. A dwarf, represented by a “☺” icon, blinking with a blue downward-pointing arrow signifies thirst, a gray capital “Z” means the dwarf is sleeping, and a variety of exclamation points suggest various levels of unrest: blue for melancholy, purple for possession, teal for insanity, and yellow for the dreaded tantrum. On top of all these codes, confusion arises when a single dwarf inhabits a space with other objects, extending a simple blink mechanic beyond two frames. In *Dwarf Fortress*, space itself behaves oddly owing to these text-based logics. As the saying goes, one tile is big enough to contain one thousand dragons lying down but too small to hold two dwarves standing up. Furthermore, one thousand dwarves lying in the same bed might render an illegible panoply of blinking status updates, and only one dwarf could stand up to leave at any one given time. Modifications that attempt to render *Dwarf Fortress* in three dimensions encounter similar visualization challenges given the spatial eccentricities of Tarn's tile-based world.
 - 4 *Dwarf Fortress* is by no means the only game in which players have attempted to create a functional computer. From *LittleBigPlanet* (2008) to *Minecraft* (2010–), students and hobbyists interested in computer engineering have transformed various game platforms into constrained development environments. Essentially, any game that has the capacity to produce a logic gate also has the potential to be transformed into a computer. W. Daniel Hillis's (1998) tinker-toy computer is proof of this basic concept, which he elaborates in his

- aptly named book *The Pattern on the Stone*. Hillis writes, “Present-day computers are built of transistors and wires, but they could just as well be built, according to the same principles, from valves and water pipes, or from sticks and strings” (viii). In addition to the Dwarven Computer, there have been experiments such as the fully functional, 16-bit ALU in *Minecraft* and the “Little Big Calculator” in *LittleBigPlanet*. These games continue to function as platforms, strange IDEs for playing with programming and programming play.
- 5 It is telling that Tarn Adams (2009) explicitly invokes J. R. R. Tolkien’s *The Silmarillion*, the encyclopedia of mythic short stories, genealogies, maps, etymologies, and songs that form the “back end” to the more narratively cohesive *Lord of the Rings* (the far more commonly cited inspiration for fantasy fiction). *The Silmarillion* was not a stable mythology but “a continuing and evolving creation extending over more than half a century” (Tolkien 1977, 7). It was the database Tolkien used to build his world and on which he subsequently relied to write his more popular novels. Tarn Adams (2009) has also noted the influence of *Romance of the Three Kingdoms* and *Water Margin* on his work. In *Water Margin*, for example, there are one hundred and eight heroes, and *Romance of the Three Kingdoms* chronicles the lives of hundreds of characters. These texts resonate with the scope, scale, and dynamics of *Dwarf Fortress*.
 - 6 Wardrip-Fruin (2009, 15) contrasts the *Tale-Spin* effect with what he calls the “*Eliza* effect.” *Eliza* (1966), also known as *Doctor*, was one of the first chat bots. The program simulated conversation with a Rogerian psychotherapist in the form of natural language and even responded to the user’s input. Although the mechanics of *Eliza-Doctor* are deceptively simple in the way they convert a user’s answers into questions, the expectations of the audience made the AI appear more complex than it actually was. *Dwarf Fortress* suffers from the opposite affliction to the extent that players of the game resorted to building their own *Dwarven Therapist* (2009–). This software helps players to “interpret” the dwarves’ complex moods, personalities, and emotional capacities, dispelling the *Tale-Spin* effect by transforming obscure, textual data into straightforward spreadsheets in line with the more traditional functions of a score sheet or status menu.
 - 7 The associations of the terms *prehistoric* and *posthistoric* reinforce this point. The term *prehistoric* generally refers to the time before the emergence of writing as an inscription technology. As Kittler (1999, 4) notes, “history was the homogenized field that, as an academic subject, only took account of literate cultures. Mouths and graphisms

were relegated to prehistory.” What is sometimes referred to as the posthistorical condition of new media also reflects the displacement of print as the privileged form of historical inscription. There are both technological and economic reasons for why new media have been accused of reflecting a dehistoricized impulse. Wendy Chun (2008) describes digital storage as a condition of “enduring ephemeral,” and Terry Harpold (2009) has argued that the history of networked and programmable media has been subject to a dehistoricized “upgrade path”—the endlessly rewritable “new” in a new media economically constructed around planned obsolescence.

REFERENCES

- Adams, Tarn, with Kyle [Capntastic] Brodzky and Nathan [Rainseeker]. 2009. “Dwarf Fortress Talk #1.” Bay 12 Games. August 6. http://www.bay12games.com/dwarves/df_talk.html.
- Adams, Tarn, and Zach Adams. 2006. “Battlefield Lunch.” Bay 12 Games. May 31. http://www.bay12games.com/dwarves/story/tt_battlefield_lunch.html.
- . 2006–11. “Threetoe’s Stories, and Analysis.” Bay 12 Games. http://www.bay12games.com/dwarves/dev_story.html.
- Apple. 2012. “Mac 101: Time Machine.” June 18. <http://support.apple.com/kb/HT1427>.
- Boluk, Stephanie, and Patrick LeMieux. 2012. “Play: Strange IDEs and the Design for a Dwarven Computer.” Critical Codes Studies Working Group 2012. February 16. http://wg12.criticalcodestudies.com/discussion/54/play-strange-ides-and-the-design-for-a-dwarven-computer#Item_10.
- Chun, Wendy Hui Kyong. 2008. “The Enduring Ephemeral, or the Future Is a Memory.” *Critical Inquiry* 35, no. 1: 148–71.
- Costikyan, Greg. 2011. “Dwarf Fortress: A Game from a Parallel (and Better?) Universe.” <http://playthisthing.com/dwarf-fortress>.
- Deacon, Terrence William. 2012. *Incomplete Nature: How Mind Emerged from Matter*. New York: W. W. Norton.
- De Landa, Manuel. 1995. *The Geology of Morals: A Neo-materialist Interpretation*. <http://www.t0.or.at/delanda/geology.htm>.
- . 1997. *A Thousand Years of Nonlinear History*. New York: Zone Books.
- Ganapati, Priya. 2010. “Geeky Gamers Build Working Computers Out of Virtual Blocks.” October 12. <http://www.wired.com/gadgetlab/2010/10/virtual-computers/>.

- Harpold, Terry. 2009. *Ex-foliations: Reading Machines and the Upgrade Path*. Minneapolis: University of Minnesota Press.
- Hillis, W. Daniel. 1998. *The Pattern on the Stone: The Simple Ideas That Make Computers Work*. New York: Basic Books.
- Jameson, Fredric. 2005. *Archaeologies of the Future: The Desire Called Utopia and Other Science Fictions*. New York: Verso.
- Johnson, Neil, Guannan Zhao, Eric Hunsader, Jing Meng, Amith Ravindar, Spencer Carran, and Brian Tivnan. 2012. "Financial Black Swans Driven by Ultrafast Machine Ecology." February 7. <http://arxiv.org/abs/1202.1448>.
- Kittler, Friedrich A. 1999. *Gramophone, Film, Typewriter*. Translated by Geoffrey Winthrop-Young and Michael Wutz. Stanford, Calif.: Stanford University Press.
- Montfort, Nick. 2010. "Computing with Your Torch." *Post Position*. September 30. <http://nickm.com/post/2010/09/computing-with-your-torch/>.
- Suits, Bernard. (1978) 2005. *The Grasshopper: Games, Life, and Utopia*. Peterborough, Ont.: Broadview Press.
- Tolkein, Christopher. 1977. Foreword to *The Silmarillion*. Boston: Houghton Mifflin.
- Tyson, Peter. 2012. *Getting Started with Dwarf Fortress: Learn to Play the Most Complex Video Game Ever Made*. Sebastopol, Calif.: Oreilly.
- Various. 2006–7. "Dwarf Fortress—Boatmurdered: Part 18 by StarkRaving Mad." Let's Play Archive. April 14. <http://lparchive.org/Dwarf-Fortress-Boatmurdered/Update%201-17/>.
- Wardrip-Fruin, Noah. 2009. *Expressive Processing: Digital Fictions, Computer Games, and Software Studies*. Cambridge, Mass.: MIT Press.
- Weiner, Jonah. 2011. "The Brilliance of Dwarf Fortress." *New York Times*, July 21. <http://www.nytimes.com/2011/07/24/magazine/the-brilliance-of-dwarf-fortress.html>.
- White, Hayden V. 1987. *The Content of the Form: Narrative Discourse and Historical Representation*. Baltimore: Johns Hopkins University Press.